

**AWS – Amazon Cloud Foundations Labs 4-6**

Brendon Ahn | Advanced Cisco | 2022-2023



**Purpose**

In AWS Amazon Cloud Foundation complete labs 4-6 and work with and learn more about EC2 Instances, things like EBS, CloudWatch, Creating Snapshots, etc.

**Background**

Amazon EC2 or Elastic Compute Cloud is an AWS service that we learn about in the Amazon Cloud Foundations lab and course. In labs 4-6 we explore this aspect of AWS. The things in EC2 of AWS we explore include Elastic Block Store(EBS) which is a storage service, as well as snapshots and troubleshooting common EC2 issues.

In Lab 4 we configure an EC2 Instance and learn how to monitor the EC2 instance remotely. This is really useful to companies as they can view the Cloud from pretty much anywhere. With the services EC2 has it saves a lot of time money and effort for businesses.

In Lab 5 we look EBS Volume and how to attach and detach EBS from EC2. We will also create a snapshot of the EBS. EBS is kind of like an external hard drive you can attach and death from EC2. This is really useful when you want to add additional storage to your EC2 Instances. This is an example of the scalability that is provided by AWS services which is useful for pretty much any company, especially a company like Boeing which is constantly has more and more data.

In Lab 6 we look at troubleshooting common EC2 Issues by looking at logs, snapshots and an AWS service called Amazon CloudWatch. Amazon CloudWatch is basically a metrics looker. You can look at applications, infrastructure, etc. It also features some automation which is really useful for any business as it prevents constantly looking at the metrics and instead having alarms and automated actions to resolve issues.

**Summary**

We look at EC2 more in depth and go into its features like CloudWatch, EBS, and Remote accessing, among other things. We learn to create snapshots, check logs, and add storage on EC2, furthering our Amazon Cloud Foundations and knowledge of AWS.

**Lab Procedure**

**Lab 4**

1. From the AWS Management Console go to Services and select EC2
2. Select Instances and select Lab
3. Select Volumes then Create Volume and configure

A screenshot of a computer

Description automatically generated

1. Create the Volume by clicking Create Volume(you may need to refresh to see the new volume)

A screenshot of a computer

Description automatically generated with medium confidence

1. Select My Volume

A screenshot of a computer

Description automatically generated with medium confidence

1. Go to Actions then click Attach Volume
2. Choose the Lab instance
3. Click Attach Volume

A screenshot of a computer

Description automatically generated

1. Review the details, click the Details drop-down and select show
2. Download PPK and save Labsuser.ppk
3. X out of Details
4. Download Putty and open

A screenshot of a computer

Description automatically generated

1. Click on Connection and set Seconds Between Keepalives to 30
2. Login as ec2-user
3. Input lsblk command in the Putty Session
4. Then sudo mkfs -t ext4 /dev/xvdf
5. Then input the sudo mkdir /data
6. Then sudo mount /dev/xvdf /data
7. Then verify by inputting df -h
8. In AWS Management go to Services and click EC2
9. Click Volumes and select My Volume
10. Go to Actions and select Create Snapshot

A screenshot of a computer

Description automatically generated with medium confidence

1. Enter My Snapshot as the Name and then click Create Snapshot
2. Click Snapshots

**Lab 5**

1. Open the AWS Management Console and go to Services then click VPC
2. Go to Security Groups
3. Click Create and configure as per instructions
4. Go to Inbound Rules and Add Rule, add a rule to allow access from Web Security group
5. Click on Create security group
6. Go to Services and click RDS
7. Click Subnet Groups
8. Select Create DB Subnet Group and configure

A screenshot of a computer

Description automatically generated with medium confidence

1. Go to Add Subnets and expand then under select the first two zones

A screenshot of a group

Description automatically generated with medium confidence

1. Expand the values under subnet and select subnets linked with the CIDR ranges from 10.0.1.0/24 and 10.0.3.0/24 and click Create
2. In the AWS Management go to Databases
3. Click Create Database
4. Go to Engine Options and select MySQL and choose the Dev/Test
5. Under Availability and Durability select Multi-AZ DB Instance and configure the settings

A screenshot of a computer

Description automatically generated with medium confidence

1. Under DB Instance Class select Burstable Classes and select db.t3.micro

A screenshot of a computer

Description automatically generated with medium confidence

1. Under Storage select General Purpose (SSD) and select 20 GB
2. Under Connectivity select Lab VPC
3. Under Existing VPC Security Groups select DB Security Group and unselect the default

A screenshot of a computer security group

Description automatically generated with low confidence

1. Click Additional Configuration and configure
2. Create Database then click the Lab-db link

**Lab 6**

1. From AWS Management Console go to Services and select EC2
2. Go to Web Server 1 and Actions then click on Image and Templates
3. Click Create Image then configure and click Create Image again
4. Go to Target Groups then Create Target Group then configure then click Next then Create Target Group

A screenshot of a computer

Description automatically generated with low confidence

1. Go to Load Balancers and click Create Load Balancer
2. Configure and then Create
3. Click Create Launch Configuration and configure
4. Choose the Web Server AMI then select Instance Type t3.micro

A screenshot of a computer

Description automatically generated with medium confidence

1. Select Enable EC2
2. Go to Security Groups and configure and click Select An Existing Security Group and pick Web Security Group

A screenshot of a computer

Description automatically generated with medium confidence

1. Go to Key Pair and configure
2. Click Create Launch Configuration
3. Check the LabConfig Launch Configuration box
4. Go to Actions then click Create Auto Scaling Group the name should be Labu Auto Scaling Group then click Next
5. Go to Network and configure Network: Lab VPC and select Private Subnets then click Next
6. Go to Load Balancing then click Attach To An Existing Load Balancer and select Lab Group

A screenshot of a phone

Description automatically generated with low confidence

1. Go to Additional Settings and Enable Group Metrics Collection for CloudWatch and then click Next
2. Configure Group Size

A screenshot of a phone

Description automatically generated with low confidence

1. Go to Scaling Policies and configure

A screenshot of a computer screen

Description automatically generated with low confidence

1. Go to Instances and refresh if you are not seeing Lab Instances
2. Click Target Groups and select LabGroup
3. Go to Targets
4. Refresh until the Instances say Healthy
5. Click Load Balancers and copy the DNS Name
6. In the left navigation pane, click Load Balancers
7. Go to a Web Browser and paste the DNS Name
8. Go to the AWS Management Console and got to Services and click CloudWatch
9. Select All Alarms

If not visible follow next steps

1. Go to Services and select EC2
2. Click Auto Scaling Groups and select Lab Auto Scaling Group
3. Click Automatic Scaling and pick LabScalingPolicy
4. Go to Actions then select Edit
5. Change Target Value to 50 then click Update
6. Go to Service then click on CloudWatch
7. Select All Alarms then click OK AlarmHigh

A screenshot of a computer

Description automatically generated with medium confidence

1. Refresh until it shows
2. Go back to the Web Browser and hit Load Test
3. Go to the tab with CloudWatch
4. Wait until AlarmLow goes OK and AlarmHigh is In Alarm and refresh if it isn’t updating

A screenshot of a computer

Description automatically generated with medium confidence

1. Go to Services and Select EC2
2. Click Instances

**Problems**

I didn’t realize how much you were supposed to wait so sometimes I would go back steps and verify that I did things correctly when I didn’t need to. I did at times however mess up with the Configurations with the AWS CLI and IAM which caused problems with future configuration. I also included something in the DNS name which I was not supposed to, which caused me to go back and look for issues when there weren’t any. I also forgot to get steps when doing the lab, so I had to go back and snip images and copy steps down.

**Conclusion**

In conclusion labs 4-6 of Amazon Cloud Foundations of AWS services showed how to work with EC2 Instances. In particular the big things we learned were the about EBS, troubleshooting, and Remote Access Cloud Watch. These are very important and very useful skills to have, and we gained foundations to go further with AWS.